Integrated Distributed Description Logics (extended abstract)

Antoine Zimmermann

INRIA Rhône-Alpes

Abstract. We propose a Description-Logics-based language that extends standard DL with distributed capabilities. More precisely, it offers the possibility to formally describe the semantic relations that exist between two ontologies in a networked knowledge-based system. Contrary to Distributed Description Logics [2], it is possible to compose correspondences ($\approx$ bridge rules), while still being able to hide some of the discrepancies between ontologies. Moreover, when ontologies have no nominals, no A-Box axioms, and correspondences are restricted to cross-ontology subsumption, the satisfiability of a local ontology is not influenced by ontology alignments and other ontologies, *i.e.*, local deduction is invariant to the change of the outer system. Although we do not have a complete reasoning procedure, we provide inference rules and semantic properties, and a discussion on reasoning in this formalism.

1 Introduction

Description Logics (DL) have been widely used in knowledge-based systems and serve as the foundation for the accepted standard language of the semantic web, *viz.*, OWL [4]. However, in their basic form, DL are not so much appropriate when used in a strongly distributed environment like peer to peer systems, semantic web, or other networked, heterogeneous knowledge-based systems. In distributed environments, ontology engineers want to reuse third party ontologies or, even more, *parts of* existing ontologies, while maintaining consistency, at least in their local knowledge representation.

We offer an extension of the DL formalism (Integrated Distributed Description Logics, or IDDL) which comply with the requirements of a distributed knowledge representation. The main advantages of our approach, compared to others, are (1) its separation of local semantics (which is standard DL), and global semantics; and (2) it allows composition of ontology mappings.

First item means that it is conceptually in accordance with the notion of semantic integration: local knowledge and reasoning should not be disturbed when embedded in a network of ontologies. Several research initiative have been launched to define languages specifically adapted to these issues. Some of them are based on DL. Sect. 2 presents these formalisms, and compares them to our approach. Sect. 3 presents the syntax and semantics of our new formalism. In Sect. 4, we discuss reasoning in this formalism by listing the inference rules added to standard DL reasoning, and explain the particularities (advantages and drawbacks) of the approach.

2 Related work

In this section, we do not investigate distributed knowledge-based formalism in its generality. We focus on DL-related work. The use of DL as a basis for a semantic web representation language was envisage early, and OWL supports "imports" of ontologies from a distant server. So, to a limited extend, OWL is a language for distributed architectures. However, since the "import" statement only copies the content of the identified ontology (using its URI as a URL), it does not so well comply with the specificities of an evolving, heterogeneous environment like the Web.

[2] defines a semantics for Distributed Description Logics (DDL), based on the same idea as DFOL [3]. It is built around the idea of contextualizing knowledge. More precisely, each ontology in a system relates other ontologies to itself in a directional way, enabling an ontology to "translate" others' knowledge in its own context. It is directional because the context to which knowledge is transfered determines how things are interpreted. Technically, each local ontology has its own domain of interpretation, and a *domain relation* defines how information is translated from one ontology to the other. These relations are not necessarily symmetric. The main disadvantage, with respect to this semantics, is the impossibility to compose so-called "bridge rules", *e.g.*, if local concept C_1 of ontology O_1 is seen (from O_1's context) as a subclass of foreign concept C_2 in ontology O_2, and C_2 is seen (from O_2's context) as a subclass of concept C_3 in ontology O_3, it not possible to deduce that C_1 is a subclass of C_3 (from O_1's point of view). So relations between two foreign ontologies are not really taken into account. Only the relations between foreign ontologies and the local ontology count in the interpretation of one given context.

Another possible approach, which has good features with respect to modularity of ontologies, is found in Package-based Description Logics (P-DL) [1]. In this DL-based formalism, local ontologies (or "package") can import not just full ontologies but rather named concepts or roles from foreign ontologies. Each ontology is interpreted in a local domain, but instead of relating it to others, they simply overlap on the imported terms interpretation. So there is no difference between the interpretation of a concept from the importing and the imported ontologies. The biggest problem is that it obliges the whole network of ontologies to be quite homogeneous.

In [5], a more abstract formalism has been used to compare different approaches interpreting distributed systems. DDL corresponds to what authors called a *contextualized semantics*, and they prove that it does not comply with ontology alignment composition. P-DL would rather correspond to *simple semantics*, which is tied to homogeneous and consistent systems. Finally, they propose a third formalism that is conceptually well suited for heterogeneous ontology integration and comply with alignment composition. In the present paper, we instantiate it by giving it a concrete DL-based semantics.

3 Syntax and Semantics

A IDDL knowledge base contains two components: a family of local DL ontologies, and a family of ontology alignments. In Sect. 3.1, we define many DL constructors but the formalism is appropriate for any subset of them, *e.g.*, $\mathcal{AL}$, $\mathcal{ALC}$, $\mathcal{ALCN}$, $\mathcal{SHIQ}$, $\mathcal{SHOIN}$, *etc*.

3.1 DL Syntax and Semantics

IDDL ontologies have the same syntax and semantics as in standard DL. More precisely, a DL ontology is composed of concepts, roles and individuals, as well as axioms built out of these elements. A concept is either a primitive concept A, or, given concepts C, D, role R, individuals $a_1, \ldots, a_k$, and natural number n, $\perp$, $\top$, $C \sqcup D$, $C \sqcap D$, $\exists R.C$, $\forall R.C$, $\leq nR.C$, $\geq nR.C$, $\neg C$ or $\{a_1, \ldots, a_k\}$. A role is either a primitive role P, or, given roles R and S, $R \sqcup S$, $R \sqcap S$, $\neg R$, R^-, $R \circ S$ and R^+.

Interpretations are pairs $\langle \Delta^I, \cdot^I \rangle$, where Δ^I is a non-empty set (the domain of interpretation) and $\cdot^I$ is the functon of interpretation such that for all primitive concepts A, $A^I \subseteq \Delta^I$, for all primitive roles P, $P^I \subseteq \Delta^I \times \Delta^I$, and for all individuals a, $a^I \in \Delta^I$. Interpretations of complex concepts and roles is inductively defined by $\perp^I = \emptyset$, $\top^I = \Delta^I$, $(C \sqcup D)^I = C^I \cup D^I$, $(C \sqcap D)^I = C^I \cap D^I$, $(\exists R.C)^I = \{x | \exists y. y \in C^I \wedge \langle x, y \rangle \in R^I\}$, $(\forall R.C)^I = \{x | \forall y. \langle x, y \rangle \in R^I \Rightarrow y \in C^I\}$, $(\leq nR.C)^I = \{x | \sharp\{y \in C^I | \langle x, y \rangle \in R^I\} \leq n\}$, $(\geq nR.C)^I = \{x | \sharp\{y \in C^I | \langle x, y \rangle \in R^I\} \geq n\}$, $(\neg C)^I = \Delta^I \setminus C^I$, $\{a_1, \ldots, a_k\} = \{a_1^I, \ldots, a_k^I\}$, $(R \sqcup S)^I = R^I \cup S^I$, $(R \sqcap S)^I = R^I \cap S^I$, $(\neg R)^I = (\Delta^I \times \Delta^I) \setminus R^I$, $(R^-)^I = \{\langle x, y \rangle | \langle y, x \rangle \in R^I\}$, $(R \circ S)^I = \{\langle x, y \rangle | \exists z. \langle x, z \rangle \in R^I \wedge \langle z, y \rangle \in S^I\}$ and $(R^+)^I$ is the reflexive-transitive closure of R^I.

Axioms are either subsumption $C \sqsubseteq D$, sub-role axioms $R \sqsubseteq S$, instance assertions $C(a)$, role assertions $R(a, b)$ and individual identities $a = b$, where C and D are concepts, R and S are roles, and a and b are individuals. An interpretation I *satisfies* axiom $C \sqsubseteq D$ iff $C^I \subseteq D^I$; it satisfies $R \sqsubseteq S$ iff $R^I \subseteq S^I$; it satisfies $C(a)$ iff $a^I \in C^I$; it satisfies $R(a, b)$ iff $\langle a^I, b^I \rangle \in R^I$; and it satisfies $a = b$ iff $a^I = b^I$. When I satisfies an axiom α, it is denoted by $I \models \alpha$.

An ontology O is composed of a set of terms (primitive concepts/roles and individuals) called the signature of O and denoted by $\mathrm{Sig}(O)$, and a set of axioms denoted by $\mathrm{Ax}(O)$. An interpretation I is a model of an ontology O iff for all $\alpha \in \mathrm{Ax}(O)$, $I \models \alpha$. In this case, we write $I \models O$. The set of all models of an ontology O is denoted by $\mathrm{Mod}(O)$. A semantic consequence of an ontology O is a formula α such that for all $I \in \mathrm{Mod}(O)$, $I \models \alpha$.

3.2 Distributed Systems

A Distributed System (DS) is composed of a set of ontologies, interconnected by ontology alignments. An ontology alignment describes semantic relations between two ontologies.

Syntax: An ontology alignment is composed of a set of *correspondences*. A correspondence can be seen as an axiom that asserts a relation between concepts, roles or individuals of two distinct ontologies. They are homologous to bridge rules in DDL. We use a notation similar to DDL in order to identify in which ontology a concept, role or individual is defined. If a concept/role/individual E belongs to ontology i, then we write it $i\!:\!E$. The 6 possible types of correspondences between ontologies i and j are:

Definition 1 (Correspondence). *A* correspondence *between two ontologies i and j is one of the following formula:*

- $i:C \xleftrightarrow{\sqsubseteq} j:D$ *is a cross-ontology concept subsumption;*
- $i:R \xleftrightarrow{\sqsubseteq} j:S$ *is a cross-ontology role subsumption;*
- $i:C \xleftrightarrow{\perp} j:D$ *is a cross-ontology concept disjunction;*
- $i:R \xleftrightarrow{\perp} j:S$ *is a cross-ontology role disjunction;*
- $i:a \xleftrightarrow{\in} j:C$ *is a cross-ontology membership;*
- $i:a \xleftrightarrow{=} j:b$ *is a cross-ontology identity.*

An ontology alignment is simply a set of correspondences. Together with DL ontologies, they form the components of a Distributed System in IDDL.

Definition 2 (Distributed System). *A* Distributed System *(DS for short), is a pair $\langle \mathbf{O}, \mathbf{A} \rangle$ such that $\mathbf{O}$ is a set of ontologies, and $\mathbf{A} = (A_{ij})_{i,j \in \mathbf{O}}$ is a family of alignments relating ontologies of $\mathbf{O}$.*[1]

Semantics: Distributed systems semantics depends on local semantics, but does not interfere with it. A standard DL ontology can be straightforwardly used in IDDL system. Informally, interpreting a IDDL system consists in assigning a standard DL interpretation to each local ontology, then correlating the domains of interpretation thanks to what we call an *equalizing function*.

Definition 3 (Equalizing function). *Given a family of local interpretations $\mathbf{I}$, an equalizing function ϵ is a family of functions indexed by $\mathbf{I}$ such that for all $I_i \in \mathbf{I}$, $\epsilon_i : \Delta^{I_i} \rightarrow \Delta_\epsilon$ where Δ_ϵ is called the global domain of interpretation of ϵ.*

A distributed interpretation assigns a standard DL interpretation to each ontology in the system, as well as an equalizing function that correlate local knowledge into a global domain of interpretation.

Definition 4 (Distributed interpretation). *Let $S = \langle \mathbf{O}, \mathbf{A} \rangle$ be a DS. A* distributed interpretation *of S is a pair $\langle \mathbf{I}, \epsilon \rangle$ where $\mathbf{I}$ is a family of interpretations indexed by $\mathbf{O}$, ϵ is an equalizing function for $\mathbf{I}$, such that for all $i \in \mathbf{O}$, I_i interprets i and $\epsilon_i : \Delta^{I_i} \rightarrow \Delta_\epsilon$.*

While local satisfiability is the same as standard DL, correspondence satisfaction involves the equalizing function.

Definition 5 (Satisfaction of a correspondence). *Let S be a DS, and i, j two ontologies of S. Let $\mathcal{I} = \langle \mathbf{I}, \epsilon \rangle$ be a distributed interpretation. We define satisfaction of a correspondence c (denoted by $\mathcal{I} \models_d c$) as follows:*

- $\mathcal{I} \models_d i:C \xleftrightarrow{\sqsubseteq} j:D$ *iff* $\epsilon_i(C^{I_i}) \subseteq \epsilon_j(D^{I_j})$;
- $\mathcal{I} \models_d i:R \xleftrightarrow{\sqsubseteq} j:S$ *iff* $\epsilon_i(R^{I_i}) \subseteq \epsilon_j(S^{I_j})$;
- $\mathcal{I} \models_d i:C \xleftrightarrow{\perp} j:D$ *iff* $\epsilon_i(C^{I_i}) \cap \epsilon_j(D^{I_j}) = \emptyset$;
- $\mathcal{I} \models_d i:R \xleftrightarrow{\perp} j:S$ *iff* $\epsilon_i(R^{I_i}) \cap \epsilon_j(S^{I_j}) = \emptyset$;

[1] We consistently use bold face to denote a mathematical family of elements. So, $\mathbf{O}$ denotes $(O_i)_{i \in I}$ where I is a set of indices.

- $\mathcal{I} \models_d i\!:\!a \xleftrightarrow{\in} j\!:\!C$ *iff* $\epsilon_i(a^{I_i}) \in \epsilon_j(C^{I_j})$;
- $\mathcal{I} \models_d i\!:\!a \xleftrightarrow{=} j\!:\!b$ *iff* $\epsilon_i(a^{I_i}) = \epsilon_j(b^{I_j})$.

Additionally, for all local formula $i\!:\!\phi$, $\mathcal{I} \models_d i\!:\!\phi$ iff $I_i \models \phi$ (*i.e.*, local satisfaction implies global satisfaction). A distributed interpretation $\mathcal{I}$ satisfies an alignment A iff it satisfies all correspondences of A (denoted by $\mathcal{I} \models_d A$) and it satisfies an ontology O_i iff it satisfies all axioms of O_i (denoted by $\mathcal{I} \models_d O_i$). When all ontologies and all alignments are satisfied, the DS is satisfied by the distributed interpretation. In which case we call this interpretation a *model* of the system.

Definition 6 (Model of a DS). *Let $S = \langle \mathbf{O}, \mathbf{A} \rangle$ be a DS. A distributed interpretation $\mathcal{I}$ is a* model *of S (denoted by $\mathcal{I} \models_d S$), iff:*

- *for all $O_i \in \mathbf{O}$, $\mathcal{I} \models_d O_i$;*
- *for all $A_{ij} \in \mathbf{A}$, $\mathcal{I} \models_d A_{ij}$.*

The set of all models of a DS is denoted by $\mathrm{Mod}(S)$. A formula α is a consequence of a DS ($S \models_d \alpha$) iff $\forall \mathcal{M} \in \mathrm{Mod}(S), \mathcal{M} \models_d \alpha$. This model-theoretic semantics offers special challenge to the reasoning infrastructure, that we discuss in next section.

4 Reasoning in IDDL

Reasoning in IDDL is a tricky task because there are two levels of interpretation, which are separated yet interdependent. Local DL inferences are valid in IDDL, but correspondences add new inference rules. Also, we show in Sect. 4.2 how to transpose knowledge of a DS into a localized ontology. However, this process does not guarantee completeness in general.

4.1 Inference Rules

Correspondences and axioms from several ontologies are used to deduce new axioms or correspondences, as shown in Fig. 1.

It is easy to prove that these rules lead to correct reasoning[2] but completeness is still under investigation. Rule 14 is a special case because it requires the introduction of a new individual that does not appear in any other axioms. This is due to the fact that DL does not provide any means to assert that there exists an unnamed instance of a concept. Rules 7, 8, 9, 10 and 11 show that it is possible to compose correspondences. Rule 15 shows that alignments can produce inconsistencies, independently of ontologies. It must be remarked that several seemingly intuitive results are not true in IDDL. For instance, $i\!:\!A \xleftrightarrow{\sqsubseteq} j\!:\!\neg B$ does not imply, nor is implied by $i\!:\!A \xleftrightarrow{\perp} j\!:\!B$, because injectivity of the equalizing function is not required. Moreover, if $i\!:\!A \xleftrightarrow{\sqsubseteq} i\!:\!B$ (resp. $i\!:\!a \xleftrightarrow{=} i\!:\!b$, resp. $i\!:\!a \xleftrightarrow{\in} i\!:\!A$), it is not possible to infer $i\!:\!A \sqsubseteq B$ (resp. $i\!:\!a = b$, resp. $i\!:\!A(a)$).

[2] See `http://www.inrialpes.fr/exmo/people/zimmer/DL2007Proof.pdf` for formal proofs.

$$\frac{i:A \sqsubseteq B}{i:A \xleftrightarrow{\sqsubseteq} i:B} \quad (1) \qquad \frac{i:a = b}{i:a \xleftrightarrow{=} i:b} \quad (2)$$

$$\frac{i:A(a)}{i:a \xleftrightarrow{\in} i:A} \quad (3) \qquad \frac{i:A \xleftrightarrow{\sqsubseteq} j:B \quad i:A' \xleftrightarrow{\sqsubseteq} j:B'}{i:A \sqcup A' \xleftrightarrow{\sqsubseteq} j:B \sqcup B'} \quad (4)$$

$$\frac{i:a \xleftrightarrow{=} j:b}{j:b \xleftrightarrow{=} i:a} \quad (5) \qquad \frac{i:A \xleftrightarrow{\perp} j:B}{j:B \xleftrightarrow{\perp} i:A} \quad (6)$$

$$\frac{i:A \xleftrightarrow{\sqsubseteq} j:B \quad j:B \xleftrightarrow{\sqsubseteq} k:C}{i:A \xleftrightarrow{\sqsubseteq} k:C} \quad (7) \qquad \frac{i:a \xleftrightarrow{=} j:b \quad j:b \xleftrightarrow{=} k:c}{i:a \xleftrightarrow{=} k:c} \quad (8)$$

$$\frac{i:a \xleftrightarrow{\in} j:B \quad j:B \xleftrightarrow{\sqsubseteq} k:C}{i:a \xleftrightarrow{\in} k:C} \quad (9) \qquad \frac{i:a \xleftrightarrow{=} j:b \quad j:b \xleftrightarrow{\in} k:C}{i:a \xleftrightarrow{\in} k:C} \quad (10)$$

$$\frac{i:A \xleftrightarrow{\sqsubseteq} j:B \quad j:B \xleftrightarrow{\perp} k:C}{i:A \xleftrightarrow{\perp} k:C} \quad (11) \qquad \frac{i:A \xleftrightarrow{\perp} j:B \quad i:A' \xleftrightarrow{\sqsubseteq} j:B}{i:A \sqsubseteq \neg A'} \quad (12)$$

$$\frac{i:A \xleftrightarrow{\perp} j:B \quad i:a \xleftrightarrow{\in} j:B}{i:\neg A(a)} \quad (13) \qquad \frac{i:a \xleftrightarrow{\in} j:B}{j:B(x) \quad i:a \xleftrightarrow{=} j:x} \quad (14)$$

$$\frac{i:a \xleftrightarrow{\in} j:B \quad j:B \xleftrightarrow{\perp} j:B}{\square} \quad (15)$$

Fig. 1. Inference rules in IDDL.

It is interesting to note that when the ontologies have no A-Box and do not use nominals, and when moreover correspondences are limited to cross-ontology subsumptions, it is not possible to deduce new local axioms with outer knowledge (only rules 1, 4 and 7 apply). Therefore, in this particular case, local reasoning is not disturbed by the surrounding DS. This is a particularly interesting feature when the network of ontologies is constantly evolving. Yet, it does not mean that embedding an ontology in a DS does not provide interesting knowledge. Indeed, the global semantics level is very much influenced, and this is crucial in ontology integration. Next section shows how a DS can be used to define a new DL ontology that integrate the knowledge from all the system.

4.2 Localizing Distributed Knowledge

Here, we show how to interpret a DS as a standard DL ontology, by building a standard DL interpretation out of a distributed one. The multiple signatures of the DS ontologies can be gathered into one vocabulary in the following way.

Definition 7. *The* integrated vocabulary V_S *of a distributed system* $S = \langle \mathbf{O}, \mathbf{A} \rangle$ *is defined as follows:*

– *for all primitive concept (resp. primitive role, resp. individual)* X_i *in ontology* O_i*, there exists a primitive concept (resp. primitive role, resp. individual)* $X_i^{\rightarrow}$ *in* V_S*;*

– *for all constructed concept (resp. constructed role) X_i of ontology O_i appearing in the axioms or alignments of S, there exists a **primitive** concept (resp. **primitive** role) $X_i^{\rightarrow}$ in V_S.*

The integrated vocabulary can be interpreted by a standard DL interpretation. An integrated interpretation is a specific DL interpretation defined out of a given distributed interpretation. Informally, it can be described as the mathematical composition of the local interpretation functions and the equalizing function.

Definition 8. *Given a distributed interpretation $\mathcal{I} = \langle \mathbf{I}, \epsilon \rangle$ of a system S, the* integrated interpretation *$\mathcal{I}^{\rightarrow}$ built out of $\mathcal{I}$ is a DL interpretation of V_S defined as follows:*

– *for all primitive concept $C_i^{\rightarrow}$ of V_S, $(C_i^{\rightarrow})^{\mathcal{I}^{\rightarrow}} = \{\epsilon_i(x); x \in C_i^{I_i}\}$;*
– *for all primitive role $C_i^{\rightarrow}$ of V_S, $(R_i^{\rightarrow})^{\mathcal{I}^{\rightarrow}} = \{\langle \epsilon_i(x), \epsilon_i(y)\rangle; \langle x, y\rangle \in R_i^{I_i}\}$;*
– *for all individual $a_i^{\rightarrow}$ of V_S, $(a_i^{\rightarrow})^{\mathcal{I}^{\rightarrow}} = \epsilon_i(a_i^{I_i})$.*

Of course, this interpretation can be extended to interpret constructed concepts like $\exists R^{\rightarrow}.C^{\rightarrow}$. Be careful not to confuse complex concept $\exists R^{\rightarrow}.C^{\rightarrow}$ and the *primitive* concept $(\exists R.C)^{\rightarrow}$. See Sect. 4 for details.

Deduction in the integrated vocabulary: Since integrated interpretations are standard DL interpretations, they may satisfy DL axioms over the integrated vocabulary. Theo. 1 shows how distributed satisfaction influence integrated interpretation satisfaction.

Theorem 1. *Let $\mathcal{I} = \langle \mathbf{I}, \epsilon \rangle$ be a distributed interpretation of a DS which contains concepts C_i, D_i, roles R_i, S_i, individuals $a_i, b_i, o_1, \ldots, o_n$ in ontology O_i and concept C_j, role R_j, individual a_j in ontology O_j.*

$$\begin{array}{ll}
I_i \models i:C_i(a_i) \Longrightarrow \mathcal{I}^{\rightarrow} \models C_i^{\rightarrow}(a_i^{\rightarrow}) & \mathcal{I} \models i:C_i \xleftrightarrow{\sqsubseteq} j:C_j \Longrightarrow \mathcal{I}^{\rightarrow} \models C_i^{\rightarrow} \sqsubseteq C_j^{\rightarrow} \\
I_i \models i:R_i(a_i, b_i) \Longrightarrow \mathcal{I}^{\rightarrow} \models R_i^{\rightarrow}(a_i^{\rightarrow}, b_i^{\rightarrow}) & \mathcal{I} \models i:R_i \xleftrightarrow{\sqsubseteq} j:R_j \Longrightarrow \mathcal{I}^{\rightarrow} \models R_i^{\rightarrow} \sqsubseteq R_j^{\rightarrow} \\
I_i \models i:C_i \sqsubseteq D_i \Longrightarrow \mathcal{I}^{\rightarrow} \models C_i^{\rightarrow} \sqsubseteq D_i^{\rightarrow} & \mathcal{I} \models i:C_i \xleftrightarrow{\perp} j:C_j \Longrightarrow \mathcal{I}^{\rightarrow} \models C_i^{\rightarrow} \sqsubseteq \neg(C_j^{\rightarrow}) \\
I_i \models i:a_i = b_i \Longrightarrow \mathcal{I}^{\rightarrow} \models a_i^{\rightarrow} = b_i^{\rightarrow} & \mathcal{I} \models i:R_i \xleftrightarrow{\perp} j:R_j \Longrightarrow \mathcal{I}^{\rightarrow} \models R_i^{\rightarrow} \sqsubseteq \neg(R_j^{\rightarrow}) \\
\mathcal{I} \models i:a_i \xleftrightarrow{=} j:b_j \Longrightarrow \mathcal{I}^{\rightarrow} \models a_i^{\rightarrow} = a_j^{\rightarrow} & \mathcal{I} \models i:a_i \xleftrightarrow{\in} j:C_j \Longrightarrow \mathcal{I}^{\rightarrow} \models C_j^{\rightarrow}(a_i^{\rightarrow})
\end{array}$$

Moreover, the following assertions hold:

$$\begin{array}{ll}
\mathcal{I}^{\rightarrow} \models C_i^{\rightarrow} \sqcup D_i^{\rightarrow} \sqsubseteq (C_i \sqcup D_i)^{\rightarrow} & \mathcal{I}^{\rightarrow} \models R_i^{\rightarrow} \sqcup S_i^{\rightarrow} \sqsubseteq (R_i \sqcup S_i)^{\rightarrow} \\
\mathcal{I}^{\rightarrow} \models (C_i \sqcup D_i)^{\rightarrow} \sqsubseteq C_i^{\rightarrow} \sqcup D_i^{\rightarrow} & \mathcal{I}^{\rightarrow} \models (R_i \sqcup S_i)^{\rightarrow} \sqsubseteq R_i^{\rightarrow} \sqcup S_i^{\rightarrow} \\
\mathcal{I}^{\rightarrow} \models C_i^{\rightarrow} \sqcap D_i^{\rightarrow} \sqsubseteq (C_i \sqcap D_i)^{\rightarrow} & \mathcal{I}^{\rightarrow} \models R_i^{\rightarrow} \sqcap S_i^{\rightarrow} \sqsubseteq (R_i \sqcap S_i)^{\rightarrow} \\
\mathcal{I}^{\rightarrow} \models (\exists R_i.C_i)^{\rightarrow} \sqsubseteq \exists(R_i^{\rightarrow}).(C_i^{\rightarrow}) & \mathcal{I}^{\rightarrow} \models (R_i^{-})^{\rightarrow} \sqsubseteq (R_i^{\rightarrow})^{-} \\
\mathcal{I}^{\rightarrow} \models \exists R_i^{\rightarrow}.\top \sqsubseteq (\exists R_i.\top)^{\rightarrow} & \mathcal{I}^{\rightarrow} \models (R_i^{\rightarrow})^{-} \sqsubseteq (R_i^{-})^{\rightarrow} \\
\mathcal{I}^{\rightarrow} \models (\{o_1, \ldots, o_n\})^{\rightarrow} \sqsubseteq \{o_1^{\rightarrow}, \ldots, o_n^{\rightarrow}\} & \mathcal{I}^{\rightarrow} \models (R_i^{+})^{\rightarrow} \sqsubseteq (R_i^{\rightarrow})^{+} \\
\mathcal{I}^{\rightarrow} \models \{o_1^{\rightarrow}, \ldots, o_n^{\rightarrow}\} \sqsubseteq \{(o_1, \ldots, o_n\})^{\rightarrow} & \mathcal{I}^{\rightarrow} \models (R_i \circ S_i)^{\rightarrow} \sqsubseteq R_i^{\rightarrow} \circ S_i^{\rightarrow}
\end{array}$$

Each of the previous assertions is quite easy to prove, but fastidious. Therefore, we do not reproduce them here but interested readers can find them online.[3] For all other

[3] `http://www.inrialpes.fr/exmo/people/zimmer/DL2007Proof.pdf`

constructors and subsumptions, counter examples can be found where the interpretation does not satisfy them. This can also be found online.

This theorem allows compiling axioms that are satisfied by all the models of a DS. Therefore, it is possible to build a new ontology that integrate knowledge from distributed ontologies and alignments. Such an ontology correctly represents knowledge of the system, but might not be complete. Nonetheless, Theo. 1 can be used as the basis of a compilation algorithm which integrate aligned ontologies in a modular way. The compiled ontology itself could be embedded in a distributed system.

5 Conclusion and further work

We have proposed a new formalism for distributed systems composed of ontologies and ontology alignments. Its semantics has the advantage of being able to compose correspondences (*i.e.*, to deduce a new alignment from a chain of alignments exists from the first to the second ontology). Given some restrictions, it also offers strong robustness, since the absence of A-Box and nominals, together with only cross-ontology subsumption correspondences, guarantee that local deduction is invariant to the change of the outer system (*i.e.*, alignments and other ontologies). Finally, it seems to be a good candidate semantics for ontology integration and modularization, because of its two-level semantics.

However, it still needs theoretical investigation. The most important work in the continuity of what is proposed here is the design of a deduction procedure. This paper already provides correct deductive rules, but completeness is not guaranteed in the general case. Developing a tableau-like algorithm is hard because the two levels of semantics interact with each others, although they are not processable together by usual methods. Such a procedure would open the way to implementation and tests.

References

1. J. Bao, D. Caragea, and V. Honavar. On the semantics of linking and importing in modular ontologies. In I. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold, and L. Aroyo, editors, *The Semantic Web - ISWC 2006, 5th International Semantic Web Conference*, volume 4273 of *LNCS*, pages 72–86. Springer, November 2006.
2. A. Borgida and L. Serafini. Distributed Description Logics: Directed domain correspondences in federated information sources. In R. Meersman and Z. Tari, editors, *On the Move to Meaningful Internet Systems 2002: Confederated International Conferences CoopIS, DOA, and ODBASE 2002*, volume 2519 of *LNCS*. Springer, 2002.
3. C. Ghidini and L. Serafini. Distributed First Order Logics. In Dov M. Gabbay and Maarten de Rijke, editors, *Frontiers of Combining Systems 2*, volume 7 of *Studies in Logic and Computation*, pages 121–139, Baldock, Hertfordshire, UK, 2000. Research Studies Press.
4. P. Patel-Schneider, P. Hayes, and I. Horrocks. OWL Web Ontology Language Semantics and Abstract Syntax. Technical report, World Wide Web Consortium (W3C), 2004.
5. A. Zimmermann and J. Euzenat. Three semantics for distributed systems and their relations with alignment composition. In I. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold, and L. Aroyo, editors, *The Semantic Web - ISWC 2006, 5th International Semantic Web Conference*, volume 4273 of *LNCS*, pages 16–29. Springer, November 2006.